RSAC 2025 Conference

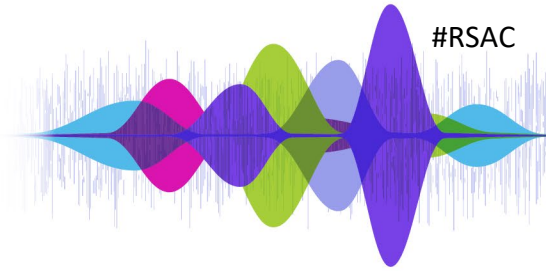
Many Voices.
One Community.

SESSION ID: IAIS-R03

Red Teaming AI:
50 years of failure, but this time, for sure!

Tanya Janca
Staff DevRel
Semgrep

Adam Shostack
Founder
Shostack + Associates



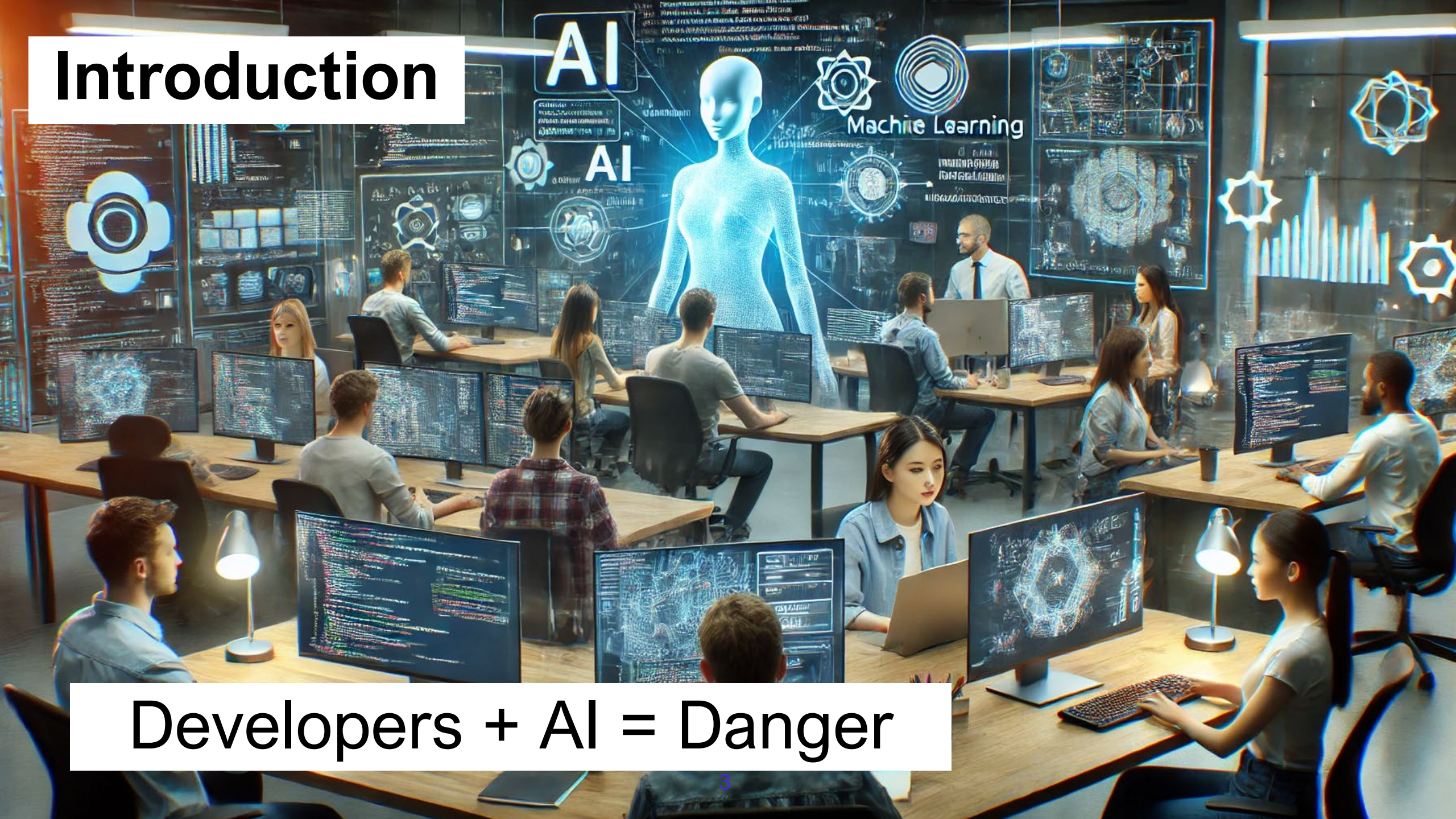
Disclaimer

Presentations are intended for educational purposes only and do not replace independent professional judgment. Statements of fact and opinions expressed are those of the presenters individually and, unless expressly stated to the contrary, are not the opinion or position of RSA Conference LLC or any other co-sponsors. RSA Conference LLC does not endorse or approve, and assumes no responsibility for, the content, accuracy or completeness of the information presented.

Attendees should note that sessions may be audio- or video-recorded and may be published in various media, including print, audio and video formats without further notice. The presentation template and any media capture are subject to copyright protection.

© 2025 RSA Conference LLC or its affiliates. The RSAC and RSAC CONFERENCE logos and other trademarks are proprietary. All rights reserved.

Introduction



Developers + AI = Danger

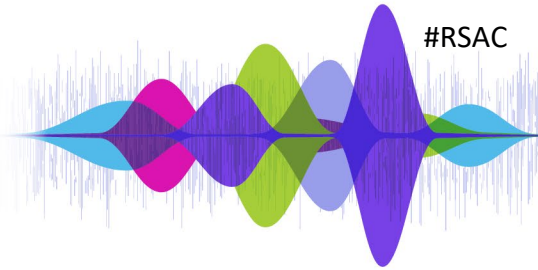
Introduction

Red Teaming has not solved the problem

Theory

Threat modeling can reduce the risk of AI.

What is threat modeling?

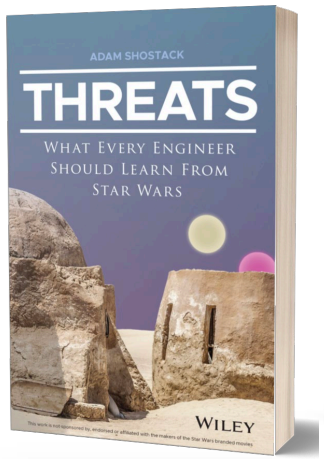


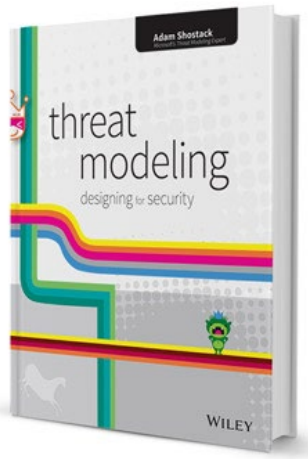
- Using models to help us think about security
- “Threats” meaning possible future problems
- TM applies to both systems you produce and systems you deploy
- The “measure twice, cut once” of security
- “You can fix it on the drawing board with an eraser, or on the job site with a sledgehammer” — Frank Lloyd Wright

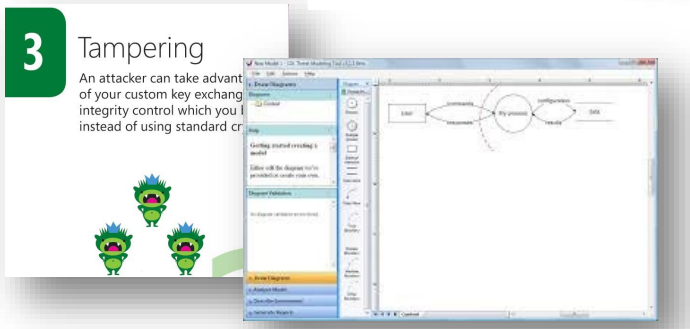
The Four Question Frame for Threat Modeling

1. What are we working on?
2. What can go wrong?
3. What are we going to do about it?
4. Did we do a good job?

About Adam Shostack







IriusRisk

CVE

black hat

W

SHOSTACK + ASSOCIATES

Semgrep

SHOSTACK + ASSOCIATES

About Tanya Janca

- Staff Developer Relations @ Semgrep
- AKA @SheHacksPurple
- Author: Alice and Bob Learn Secure Coding &

  Alice and Bob Learn Application Security
- 28+ years in tech, Sec + Dev
- Founder: We Hack Purple, OWASP DevSlop,

  #CyberMentoringMonday, WoSEC
- Advisor: Katilyst
- Faculty: IANs Research

Semgrep
SHOSTACK + ASSOCIATES

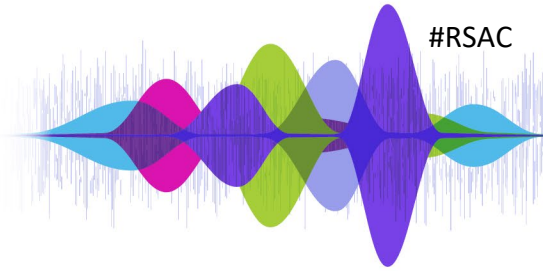
RSAC™ 2025 Conference

What are we working on when we work with AI?

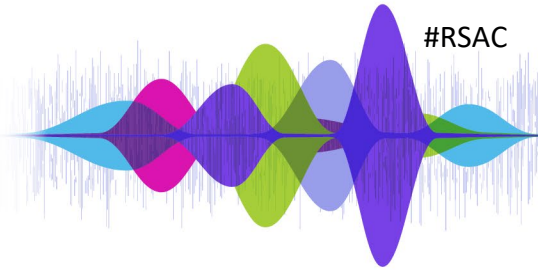
What are we working on?

- Our LLM
- Someone else's LLM (Eg, from huggingface)
- Someone else's service
  - As an LLM (eg, api.openai.com)
  - As a complete service (eg, Defender CoPilot)
- Shadow AI
- Let's talk about each!

Our LLM



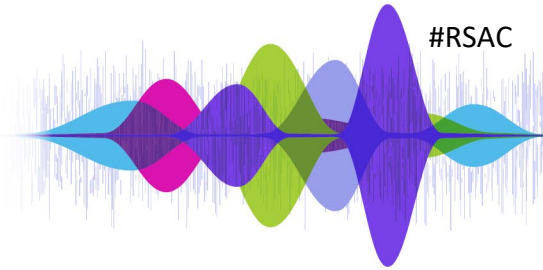
- We train and fine-tune our own LLM model using proprietary data.
  - An LLM is a statistical token predictor
  - “The prompts” drive statistics, and we get words/pix/music/etc
- We control the architecture, training process, and optimization.
- We ensure customization for our specific needs (e.g., accuracy, domain expertise, ethical considerations).



Someone Else's LLM

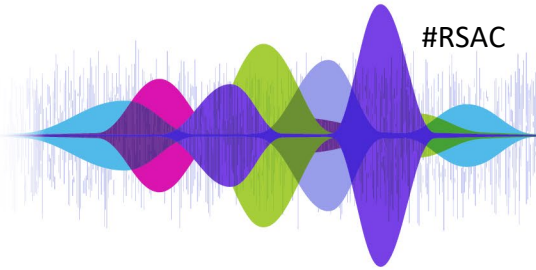
- We use someone else's model
  - Either by downloading it and running an "open source" model
  - Or by calling their API
- The provider trains and optimizes and maybe updates it
- We use prompt engineering to integrate it into our application.

Someone Else's Service



- Instead of integrating an AI model directly, we use a complete AI-driven service (e.g., AI-based fraud detection, recommendation engines, or transcription services).
- We rely on the provider's API and infrastructure.

Shadow AI (Unauthorized AI Usage)



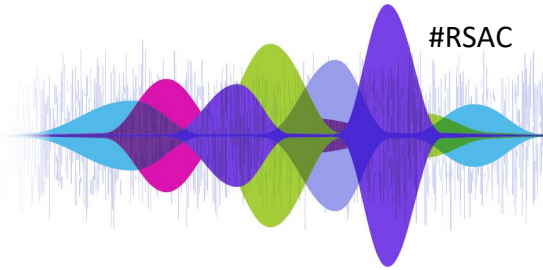
- Employees or departments use AI tools without approval or security oversight.
- Often involves free or consumer AI tools for automating tasks, writing code, or analyzing data.

RSAC™ 2025 Conference

What can go wrong when we work with AI?

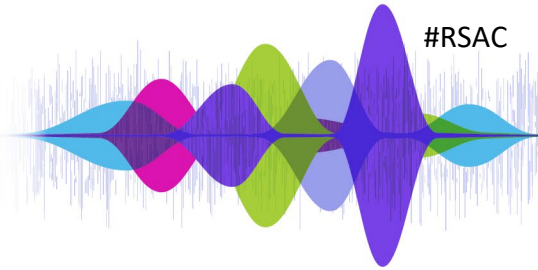
Many Voices.
One Community.

Adversarial Attacks on AI



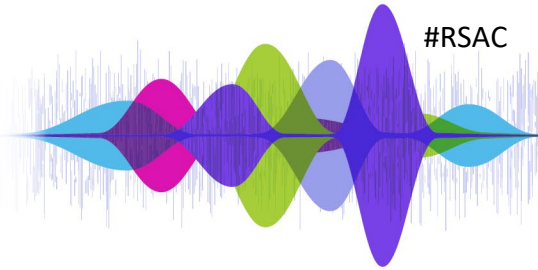
1. Model Inversion Attacks - Attackers reconstruct input data (e.g., images, private health records) by exploiting a trained model.
2. Data Poisoning Attacks- Attackers inject manipulated data into a training dataset to corrupt the AI model.
3. Adversarial - Attackers add small, unnoticeable changes to input data to fool AI models.
4. Model Extraction Attacks - Attackers query a black-box AI model to create a near-identical replica.
5. Backdoor Attacks - Attackers train a model with hidden triggers that activate under specific conditions.

Adversarial Attacks on AI – Page 2



6. Membership Inference Attacks - Attackers determine if a specific data point was used to train an AI model.
7. Trojan Attacks - Attackers implant hidden malicious behavior in an AI model during training.
8. Prompt Injection Attacks - Attackers manipulate AI chatbots by embedding malicious inputs into prompts.
9. Side-Channel Attacks - Attackers analyze indirect information (e.g., response times or power consumption) to infer sensitive data.

What can go wrong other than security?



- Bias
- Fairness
- Privacy violations
  - Memorization
  - Association

All the traditional threats

- STRIDE
- Kill Chains
- XSS
- Dependency problems
- Etc…
- (Pick your least favorite OWASP top ten)

Traditional Attacks Work

- Despite AI being magic, those traditional attacks still work
- Even in 2025 designs like MCP
  - Everything Wrong with MCP (Shrivu Shankar, April 13, 2025)
  - Jumping the line: How MCP servers can attack you before you ever use them (Trail of Bits, April 21, 2025)
  - Did no one threat model MCP?

https://blog.sshh.io/p/everything-wrong-with-mcp
https://blog.trailofbits.com/2025/04/21/jumping-the-line-how-mcp-servers-can-attack-you-before-you-ever-use-them/

All the new threats!

- Analysis from non-profits
  - Berryville Institute
  - MITRE Atlas
  - OWASP (Top Ten for LLMs, AI Exchange)
- Commercial AI Security Frameworks
  - Google
  - Snowflake
  - Databricks

What else could frameworks cover?

- Risk Management (NIST, EU AI Act)
- Privacy
- Bias
- Explainability for various audiences
- Hallucinations (the bad kind, not the expected kind?)
- Shadow AI
- Copyright

RSAC 2025 Conference

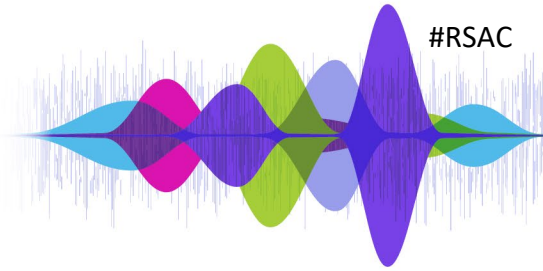
What are we going to do (with AI)?

(development, deployment, software you acquire)

Many Voices.
One Community.

Do we build an AI Program?

- Option 1: Build an AI program
  - AI specific: Security, privacy, impact, bias, trust, or more.
- Option 2: Extend an appsec program
- Tradeoff: review fatigue vs specialized advice



1. Establish an AI-SDLC

- Training
- Technology – AI specific tools, scanners, and guardrails
- Code review and/or Static Analysis
- Defensive AI frameworks (rapidly developing, check your requirements)
- Testing tools
- Software composition analysis for AI
- AI Monitoring
- License analysis

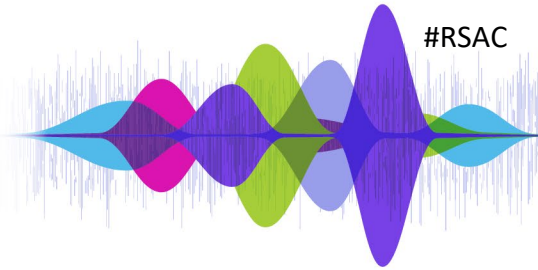
1. Establish an AI-SDLC – page 2

- Project requirements
- Threat modeling (make AI a regular TM question)
- Architecture review and documentation
- Pentesting engagements
- Using AI as part any of these is fun — and advancing very fast

2. Cross Functional Work

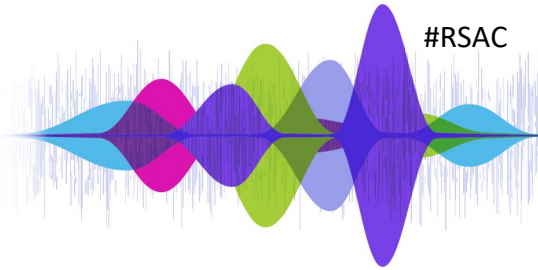
- Create a cross functional AI Task Force
  - Engineering
  - GRC
  - Legal and Compliance
  - Data Science
  - AI engineers
  - HR
  - Training for staff

3. AI Threat Monitoring & Response Plans



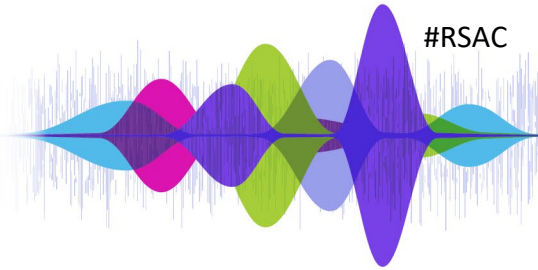
- Develop an AI Incident Response Team (IRT) to detect and respond to AI threats.
- Use AI security monitoring tools to detect anomalies.
- Implement adversarial testing in AI models before deployment.

4. Control AI Data & Privacy Risks



- Encrypt training data and enforce data anonymization.
- Restrict access to AI models—strict access control.
- Use federated learning where possible to reduce centralized data risk.

5. AI Ethics and Governance Policies



- Define AI ethics policies to govern fairness, transparency, and accountability.
- Regularly audit AI models for bias and unintended behavior.
- Implement explainability (XAI) techniques to make AI decisions more easily interpretable.

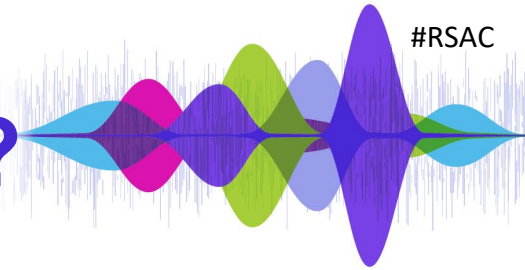
6. Reduce Shadow AI

- Implement Shadow AI policies—require IT & GRC approval for AI tools.
- Train employees on AI risks, safe AI usage, and ethical concerns.
- Monitor unauthorized AI API usage via cloud security tools.
- Scan code for unauthorized usage via SAST.

RSAC™ 2025 Conference

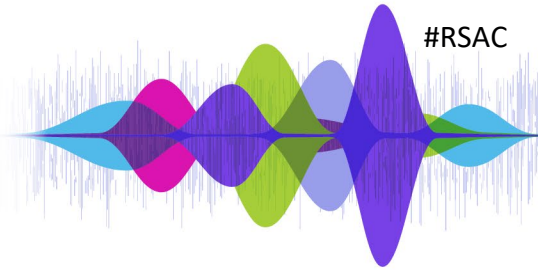
Did we do a good job?

Have you looked at what your staff is doing?



- No. Sorry, your people are outrunning you today.
- Shadow AI literally everywhere.
- Most employees have never had any training.
- Everyone is ‘winging it’.

Code and data are intermingled in LLMs



- Code/data separation failures underly
  - XSS
  - SQL Injection
  - Stack smashing
- Code/data are not separable in LLM systems
- This is a fundamental security problem
- Not going to be solved by more red teaming

Is there a clear path to solving this?

- No
- ^&*$%^*&^ hell no
- It’s fundamental to how LLMs work
  - Statistical next token prediction
  - Trained on “curated” data
- Lots of Turing awards for solving it

Industry is working on it

- For example, “Defeating Prompt Injections by Design”
- How it (CaMeL) works
  - First LLM translates the prompt into python
  - Runs code analysis on the python code
  - Provides the python to a *quarantined* LLM for execution
- (Wasn’t public when we submitted the talk)

Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, Florian Tramèr (Google Deepmind, Google, and ETH Zurich) March 24, 2025, https://arxiv.org/abs/2503.18813

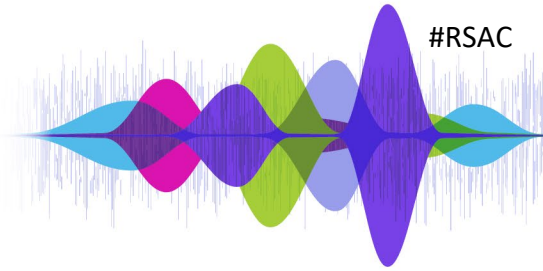
We complied with the rules! We done?

- Well, maybe
- Is it good enough for your boss?
  - Your customers?
  - Your brand?

RSAC 2025 Conference

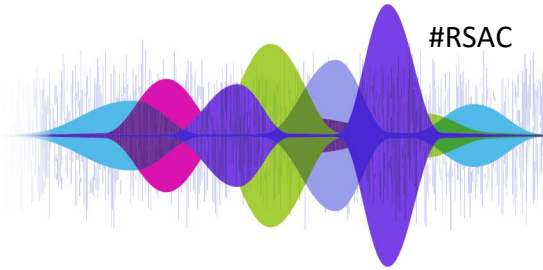
Future Challenges

Predictive analytics for anticipating AI threats and vulnerabilities



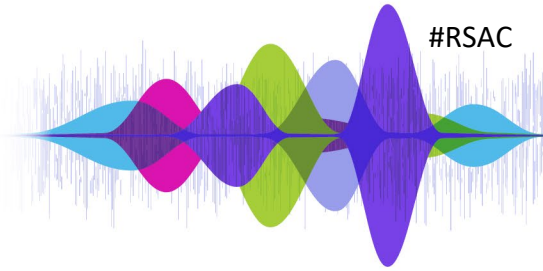
- Predictive analytics means using machine learning, statistical modeling, and historical data to identify potential threats ***before*** they materialize.
- Options for prediction:
  - Threat Intelligence & Risk Forecasting
  - Adversarial Attack Detection & Prevention
  - Early Detection of Model Drift & Data Poisoning
  - AI Model Vulnerability Prediction
  - Predictive Access Control & Insider Threat Detection
  - Automated AI Security Incident Response

Emerging technologies and practices worth watching



- AI driven ‘cyber solutions’: XDR & SOAR, threat hunting, Anomaly Detection
- AI Generated Code – but focused on more secure solutions
- Adversarial ML defenses, which use AI
- AI-Powered Software Supply Chain Security SBOM, SCA, exploitability
- AI powered vuln fixes
- AI-Resilient Identity

Emerging technologies and practices worth watching



- AI Governance & Compliance Automation
- Shadow AI & Unauthorized AI Detection
- AI-Generated Code Sandboxing
- Self-Healing AI Security Systems or code systems
- AI generated training & testing data – for better privacy
- AI-Powered Cyber Deception – use AI to trick and catch threat actors

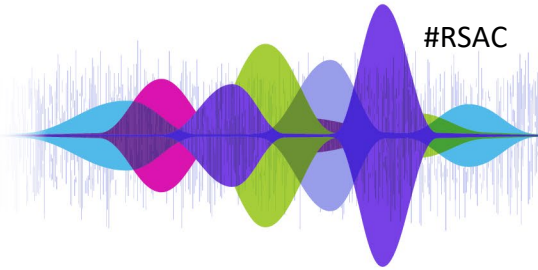
RSAC 2025 Conference

Conclusion

Apply What You Have Learned Today

- Next week you should:
  - Create a cross functional AI security task force
- In the first three months following this presentation you should:
  - Identify critical threats from AI in your org
    - Threat modeling, red teaming, reading the news and trying those things…
  - Document the threats, and plan to mitigate
- Within six months you should:
  - Include AI problems in all new threat modeling activities
  - Implement the most important mitigations, tools and practices

Apply what you have learned (2)



- Ongoing
  - A threat modeling program to anticipate AI threats
  - AI incident response to manage issues as they arrive
  - Budget and staff to manage each

- Learning + adjusting will be an ongoing part of delivering AI value

RSAC™ 2025 Conference

Many Voices.
One Community.

Adam Shostack

shostack.org

Tanya Janca

semgrep.dev